

**Amendments to the Claims:**

This listing of claims will replace all prior versions, and listings, of claims in the application:

**Listing of Claims:**

1. (currently amended): A method of error recovery of a bound remote method invocation (RMI) interface object process, the method comprising:
  - binding an interface object of a parent process with an RMI process; and
  - ~~calling an object~~ starting a monitoring agent associated with the ~~parent process~~ interface object, the ~~object initiating~~ monitoring agent comprising a thread to perform the steps of:
    - determining if the ~~RMI process~~ interface object is bound with the ~~parent RMI~~ process, wherein if the interface object is not bound with an active RMI process, an error occurs, the determining step including:
      - obtaining a bound uniform resource locator (URL) list from the RMI process; and
      - determining whether the ~~parent process's~~ interface object's name is in the bound URL list of the RMI process; and
      - rebinding the ~~parent process~~ interface object with an active RMI process when the ~~thread~~ monitoring agent determines that its ~~parent process~~ interface object is not bound with an active RMI process, thereby recovering from the error.
- 2-5. (cancelled).
6. (currently amended): The method of claim 1, comprising:
  - binding a second interface object of a second parent process with an RMI process; and
  - calling a second ~~object~~ monitoring agent associated with the ~~second parent process~~ interface object, the second ~~object initiating~~ monitoring agent comprising a second thread to perform the steps of:
    - monitoring the status of RMI processes; and
    - rebinding the second ~~parent process~~ interface object with an active RMI process when the second ~~thread~~ monitoring agent determines that the second ~~parent process~~ interface object is not bound with an active RMI process.

7. (currently amended): The method of claim 1, wherein the step of binding ~~a parent process~~ the interface object comprises:

binding one of an RMI daemon, a distributed task facility daemon, a log manager daemon, or a domain manager daemon, with an active RMI daemon.

8. (currently amended): The method of claim 1, comprising:  
terminating the thread of the monitoring agent when the parent process is terminated.

9. (currently amended): A network system, comprising:  
a plurality of remote nodes, at least one of the remote nodes running a remote method invocation (RMI) process; and

a management server for managing the remote nodes, the management server including at least one processor for running an RMI process and at least one management process, each at least one management process being associated with ~~an object capable of initiating~~ a monitoring agent comprising a thread to perform the steps of:

determining if ~~the RMI process~~ an interface object of a management process is bound with ~~the parent~~ an RMI process, wherein if the interface object is not bound with an active RMI process, an error occurs, the determining step including:

obtaining a bound uniform resource locator (URL) list from the RMI process; and

determining whether the ~~parent process's~~ interface object's name is in the bound URL list of the RMI process; and

rebinding the ~~at least one management process~~ interface object with an active RMI process when the ~~thread~~ monitoring agent determines that the ~~at least one management process~~ interface object is not bound with an active RMI process, thereby recovering from the error.

10. (original): The network system of claim 9, wherein the at least one management process comprises a plurality of management processes.

11. (original): The network system of claim 9, wherein the plurality of management processes comprise:

a distributed task facility process;  
a domain manager process; and

a log manager process.

12. (original): The network system of claim 9, wherein each of the remote nodes runs a service control manager agent process for performing server management tasks.

13. (original): The network system of claim 9, wherein the management server comprises:

a secondary storage device, the secondary storage device comprising:

a data repository;

a depot; and

a web server.

14. (original): The network system of claim 9, wherein the plurality of remote nodes are arranged into at least one node group, the network system comprising a service control manager for managing the at least one node group.

15. (currently amended): A method of error recovery of a bound remote method invocation (RMI) ~~process~~ interface object, the method comprising:

a) performing a rebind call to an RMI process to provide a network address and an ~~object~~ interface object of a parent process to the RMI process; and

b) performing an initialization call to ~~an object~~ a monitoring agent associated with the ~~parent process~~ interface object, the ~~initialization call initiating~~ monitoring agent comprising a thread, the thread performing the steps of:

1) performing a list call to an active RMI process to determine whether the ~~parent process~~ interface object is bound with the RMI process, the list call obtains a bound uniform resource locator (URL) list from the RMI process and determines whether the parent process's name is in the bound URL list of the RMI process, wherein if the interface object is not bound with an active RMI process, an error occurs;

2) performing a rebind call to an active RMI process if the ~~thread~~ monitoring agent determines that the ~~parent process~~ interface object is not bound with an active RMI process, thereby recovering from the error; and

3) repeating steps 1 and 2.

16. (original): The method of claim 15, wherein the parent process is one of a remote method invocation daemon, a distributed task facility daemon, a log manager daemon, and a domain manager daemon.

17-18. (cancelled).

19. (currently amended): The method of claim 15, comprising:  
terminating the thread of the monitoring agent when the parent process is terminated.